

REMARKS

In response to the final Office Action dated June 15, 2005, Applicant respectfully requests favorable reconsideration of the above-captioned application in view of the following remarks. Claims 1-32 remain pending in this application.

Regarding the 35 U.S.C. § 102 Rejection

Claims 1-32 were rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,785,211 to Immerman et al. (referred to below as “Immerman”). Applicant respectfully traverses this rejection for the following reasons. In addition to the following arguments, Applicant incorporates herein, by reference, the arguments set forth in the March 30, 2005 Response.

As a main thrust, Immerman is directed to a system and method for distributing a run time model to a client for execution thereat (col. 2, lines 29-31). To implement this feature, Immerman discloses a Domino Off-line Service (DOCS) 62 that provides distributed computing and remote execution of various applications (col. 5, lines 3-7). In operation, the DOLS 62 provides a way for browser users (at the client site) to utilize Domino Web application offline (col. 20, lines 26-29). More specifically, using a browser, a user can take an application offline, make changes, and synchronize those changes with the online application (col. 20, lines 29-32).

The Office Action relies principally on cols. 5 and 6 of Immerman in rejecting most of the independent claims in the above-captioned application. That portion describes a local run time model 90. At the outset, it is important to note that the local run time model 90 refers to processing performed at a client site, not the server site. See for instance, col. 2, lines 55-57, which states, "FIG. 2 is a diagram illustrating the objects

1 unbundled to a local run time model in support of an API for client side execution of
2 Notes.”

3 The local run time model 90 comprises a hierarchy of models including object
4 data store model 92, security model 96, indexing model 98, replication model 94, agent
5 workflow model 99 and mail model 97 (col. 5, lines 31-41). The data store model 92
6 specifies the level of access that users and servers have to data elements (col. 5, lines 53-
7 58), the security model 96 provides a collection of log-in credentials (col. 5, line 66), the
8 indexing model 98 provides a search index which administrators and database managers
9 may apply to databases and files (col. 6, lines 15-19), the replication model 94 provides a
10 series of rules describing how to organize and synchronize databases (col. 6, lines 25 and
11 26), the agent workflow model 99 implements the execution of an agent (col. 6, lines 42
12 and 43), and the mail model 97 provides rules for forwarding information from one
13 object data store location to another (col. 6, lines 50 and 51). In this hierarchy, “the
14 design of a parent model is a prerequisite to the design of a child model” (col. 5, lines 47
15 and 48). Immerman provides an example of what this means by stating that the “security
16 model 96 is a prerequisite to mail model 97 in the sense that mail model 97 must provide
17 for verification of the identity of users accessing mail model 97 with respect to a data
18 object” (col. 6, lines 53-57). Fig. 2 shows the specific design dependencies of the above-
19 identified models.

20 The above-described subject matter is not even remotely related to what is being
21 claimed in the present application. Consider claim 1, which is reproduced below in its
22 entirety.

23
24 1. A server system, comprising:
25 one or more computers;

an application executing on the computers to receive and process client requests; and a constraint system to constrain operation of the application according to multiple different constraints, the constraint system comprising a hierarchy of constraint layers, with each constraint layer containing a set of one or more constraints that customize operation of the application.

First, the claim is directed to a “server system.” In contrast, the cited portions of Immerman disclose a “local run time model,” deployed at the client site.

Second, claim 1 recites, in part, "a constraint system to constrain operation of the application according to multiple different constraints, the constraint system comprising a hierarchy of constraint layers, with each constraint layer containing a set of one or more constraints that customize operation of the application." Immerman discloses no such constraint system. In citing columns 5 and 6 of Immerman, the Patent Office is apparently of the opinion that the model hierarchy shown in Fig. 2 has a bearing on the claimed constraint system having a hierarchy of constraint layers. But, as summarized above, that hierarchy merely illustrates the design dependencies of different components of the local run time model 90; for example, the security model 96 is a prerequisite to mail model 97 in the sense that the mail model 97 must provide for verification of the identity of users accessing mail model 97 with respect to a data object. The models *do not* pertain to a constraint system which *constrains or customizes the operation of an application* as claimed. More specifically, the various models (92, 94, 96, 97, 98, 99) in Fig. 2 do not constrain or customize the operation of the local run time model 90, but rather the models *make up (or comprise)* the model 90 itself (see col. 5, lines 31-41). That is, the models are integral parts of the local run time model 90, rather than constraints on some other, pre-existing and independent, application.

1 In response to this argument, the Final Office states, in part, that "There is no
2 limitation in the claim on the functionality of the constraint or how the constraint
3 customizes the operation of the application and therefore Immerman meets the scope of
4 the claimed limitation 'a constraint system comprising a hierarchy of constraint layers
5 that customizes the operation of the application'" (page 11, paragraph No. 3 of the Final
6 Office Action). This position is misplaced because claim 1 *does* clearly distinguish over
7 Immerman in its present form. For instance, the organization of models shown in Fig. 2
8 of Immerman does not "customize operation of the application," according to the plain
9 meaning of the word "customize." According to one online dictionary, to "customize"
10 means to "change to suit better: to alter something in order to make it fit somebody's
11 requirements better," as in "She has customized the software to suit our needs"
12 (http://encarta.msn.com/dictionary_customize.html). As explained above, the
13 organization of models shown in Fig. 2 of Immerman controls the design dependency of
14 the models. It does not *change* an application; it just defines how the models are put
15 together.

16 The Final Office Action also cites a portion of the Immerman's abstract which
17 states:

18
19 DOLS provides a layered security model that allows flexibility for controlling access to all or part
20 of an application. The highest level of security is managed through a database access control list
21 (ACL). Further refinements within the security model provide access to specific documents, and
22 their views, forms or folders, and include read access lists, write access lists, form access lists and
23 readers and authors fields.

24
25 Immerman also discloses this subject matter in col. 5, lines 13-20.

1 First, it is pointed out that, in making the rejection, the Patent Office is combining
2 two different concepts together. As a first concept, the Office Action cites the
3 organization of modules shown in Fig. 2. As a second concept, the Office Action cites
4 the layered nature of the security module. Since these concepts do not address the same
5 subject matter, it is not clear how the Office Action is proposing that these concepts be
6 interpreted in aggregate. As it stands, therefore, the rejection appears to be internally
7 inconsistent, and thus incoherent. And this is not an isolated deficiency of the rejection,
8 but a problem that pervades the Office Action as a whole. (Consider, for instance, the
9 rejection of claim 8, which recites, in part, a "constraint resolver." The Office Action
10 identifies col. 5, line 53 to col. 6, line 10 of Immerman. But that portion discusses no
11 fewer than three different models: the object store model 92, the security model 96, and
12 the indexing model. It is not clear what model or combination of models is being
13 construed as the constraint resolver. If plural models are being combined, it is not clear
14 in what manner these models are being combined.)

15 Second, even if the cited passage (in the abstract) is applied for what it describes
16 by itself, this passage fails to disclose the subject matter recited in claim 1. Immerman's
17 multi-layered security model appears to refer to client-side functionality, not functionality
18 deployed at a "server system" as claimed.

19 Third, a security model having plural layers does not *change* an application, and
20 therefore cannot be said to "customize operation of the application," as claimed,
21 according to the plain meaning of the word "customize."

22 For at least the above reasons, the Applicant respectfully submits that Immerman
23 neither discloses nor suggests the subject matter recited in claim 1.

24
25

1 Claims 2-8 depend from claim 1, and are therefore allowable for at least this
2 reason. Moreover, each of these claims recites additional features which are not
3 disclosed in, or suggested by, Immerman.

4 For instance, claims 2-6 recite details of different constraint layers in the
5 constraint hierarchy. Namely, claim 2 recites that the hierarchy comprises a constraint
6 layer that contains legally mandated constraints to constrain operation of the application
7 according to legal principles. Claim 3 recites that the hierarchy comprises a constraint
8 layer that contains company-mandated constraints to constrain operation of the
9 application according to preferences of a company that operates the application. Claim 4
10 recites that the hierarchy comprises a constraint layer that contains customer constraints
11 to constrain operation of the application according to preferences of customers. Claim 5
12 recites that the hierarchy comprises a constraint layer that contains cultural constraints to
13 constrain operation of the application according to cultural aspects. And claim 6 recites
14 that the hierarchy comprises a constraint layer that contains end user constraints to
15 constrain operation of the application according to preferences of an end user.

16 To repeat, Immerman's Fig. 2 shows an object data store model 92, a security
17 model 96, an indexing model 98, a replication model 94, an agent workflow model 99
18 and a mail model 97. These models have no bearing on the claimed constraint layers that
19 contain *legally mandated constraints, company-mandated constraints, customer*
20 *constraints, cultural constraints and end user constraints*. Further, if, in the alternative,
21 the Office Action is interpreting the claimed constraint system as Immerman's multi-
22 layer security model, the Applicant points out that different layers of a security model
23 cannot be interpreted as the above-identified specific constraint layers.

24 In rejecting these dependent claims, the Office Action also cites col. 10, line 30 to
25 col. 11, line 28 of Immerman, as well as col. 18, line 59 to col. 19, line 62 of Immerman.

1 The excerpt in cols. 10 and 11 refers to an ID policy database 114 and an ID repository
2 database 111. The excerpt in cols. 18 and 19 refers to a configuration document 232
3 having tabbed pages for basics tab 380, services tab 384, schedule tab 390, and rules tab
4 400. This excerpt also describes an offline security policy form 410, an application page
5 238, web control 241, and various other components 246-258. First, these topics do not
6 appear to have anything to do with the constraint layers recited in claims 2-6, namely,
7 constraint layers pertaining to *legally mandated constraints, company-mandated*
8 *constraints, customer constraints, cultural constraints and end user constraints*. For
9 example, the words "legal" and "cultural" do not appear anywhere in the Immerman
10 document. Second, even if, assuming *arguendo*, that the information disclosed in these
11 excerpts had some relevance to the claimed constraint layers, Immerman does not
12 describe that this information is arranged in a *constraint hierarchy* (as recited in claim 1).

13 Dependent claim 7 recites that the constraint layers are organized within the
14 hierarchy such that a first constraint layer limits a second constraint layer but the second
15 constraint layer does not limit the first constraint layer. As described above, Immerman's
16 Fig. 2 refers to design prerequisites (that is, one component must be built for another
17 component to work properly), rather than constraints in the context of the claimed
18 invention.

19 Dependent claim 8 recites that the server system (of claim 1) further comprises a
20 constraint resolver to resolve the constraint layers so that operation of the application is
21 constrained by a sum of the constraints in the layers. Immerman discloses that the
22 different models shown in Fig. 2 comprise the local run time model 90. There is
23 absolutely no hint in Immerman that these models might conflict with each other. Hence,
24 there is likewise no suggestion that Immerman's system includes a constraint resolver to
25 cope with such conflicts.

1 Based on at least the above reasons, the Applicant respectfully submits that
2 Immerman neither discloses nor suggests the subject matter recited in dependent claims
3 2-8.

4 The remaining claims (9-32) recite subject matter that is related to various
5 permutations of the above-discussed subject matter (in claims 1-8). Therefore these
6 claims are allowable for reasons that are related to those given above. In addition, these
7 claims recite additional subject matter which is not disclosed or suggested in Immerman.

8 Consider, for instance, independent claim 9. This claim is reproduced below in its
9 entirety:

10
11 9. A server system comprising:

12 one or more computers; and

13 a multi-layer application executing on the computers to handle client requests, the multi-

14 layer application comprising:

15 a problem-solving logic layer to process the client requests according to an associated
16 problem domain, the problem-solving logic layer containing one or more execution models to
17 perform various sets of tasks when processing the client requests, the problem-solving logic layer
18 producing replies to the client requests;

19 a presentation layer to structure the replies produced by the problem-solving logic layer
20 in a manner that makes the replies presentable on various client devices; and

21 a constraint hierarchy of multiple constraint layers, each constraint layer containing a set
22 of one or more constraints that specify how the replies should be structured to customize the
23 replies for specific sets of conditions.

1 The Office Action again relies, in part, on cols. 5 and 6 of Immerman, but Fig. 2
2 of Immerman in no way discloses or suggests a problem-solving logic layer and a
3 presentation layer. Furthermore, Immerman in no way discloses or suggests a constraint
4 hierarchy of multiple constraint layers, each constraint layer containing a set of one or
5 more constraints *that specify how the replies should be structured to customize the replies*
6 *for specific sets of conditions*. The hierarchy shown in Fig. 2 is used to construct the
7 local run time model 90, not to customize replies.

8 The rejection of claim 9 also identifies col. 10, line 30 to col. 11, line 28 of
9 Immerman. This passage, in part, reads:

10
11 ID policy database 114 is a highly secure collection of security policy documents 110. It
12 is accessed by DSAPI ID generator 108 in response to a user login request on channel 307 to
13 determine the security domain of that user and determine the correct response.

14
15 While this passage mentions determining a “correct response,” this subject matter is not
16 even remotely related to constraints that specify how replies should be structured to
17 customize the replies for specific sets of conditions. To repeat, Immerman’s main thrust
18 is to download functionality to a client device to allow the client device to work offline.
19 Thus, Immerman devotes significant discussion to this download procedure. This
20 download procedure is a preparatory procedure, enabling the client device to work
21 offline; as such, this procedure does not describe the use of a server system to customize
22 replies for client devices in the manner recited in claim 9. In other words, Immerman’s
23 “correct response” has nothing to do with the customized replies recited in claim 9.

24 As another example of the deficiency of the Immerman reference, claim 10
25 (which depends on claim 9), recites that the constraint layers can be selectively added or

1 removed from the constraint hierarchy independently of other layers in the multi-layer
2 application to produce different sets of constraints. In Immerman's Fig. 2, certain models
3 are prerequisites of others, indicating that these models *cannot be added or moved*
4 *independently of each other.*

5 For at least the above exemplary (and non-exhaustive) reasons, the Applicant
6 submits that Immerman does not anticipate any of claims 1-32, and respectfully requests
7 that this rejection be withdrawn. Namely, as stated in MPEP § 2131, a claim is
8 anticipated only if each and every element as set forth in the claim is found, either
9 expressly or inherently described, in a single prior art reference. *Verdegaal Bros. v.*
10 *Union Oil Co. of California*, 2 USPQ2d 1051 (Fed. Cir. 1987). Since Immerman does
11 not set forth each and every feature, it fails to anticipate the claims under § 102.
12 Moreover, for the reasons stated above, Immerman discloses a very different system than
13 the claimed invention, and therefore also does not render the claims obvious under 35
14 U.S.C. § 103.

15 In any event, MPEP § 707.07(f) states that, "Where the applicant traverses any
16 rejection, the examiner should, if he or she repeats the rejection, take note of the
17 applicant's arguments and answer the substance of it" (page 700-119, Rev. 2, May 2004).
18 The Final Office Action did not address the majority of the points made in the March 30,
19 2005 Response, and therefore fails to comply with the policy set forth in the MPEP. If
20 the rejection is repeated, the Patent Office is respectfully requested to address each of the
21 points raised above. A non-exhaustive list of these points is presented again here in
22 summary fashion:

23 • Some of the claims are directed to a "server system." In contrast, the thrust in
24 Immerman is to download functionality to a client device, so that the *client device can*

1 *work offline.* This means that the server-centric role recited in some claims is nowhere
2 disclosed or suggested by the Immerman document.

3 • Some of the claims recite a constraint hierarchy that customizes an application.
4 In contrast, Immerman discloses a local run time model having component models that
5 have a prescribed design dependency. But these component models cannot be said to
6 customize an application in the manner claimed, given the plain meaning of the word
7 "customize."

8 • Some of the claims recite constraint layers that contain legally mandated
9 constraints, company-mandated constraints, customer constraints, cultural constraints and
10 end user constraints. Immerman discloses none of these models. The Office Action has
11 not even made clear how Immerman's models (object data store model 92, security
12 model 96, indexing model 98, replication model 94, agent workflow model 99, and mail
13 model 97) are being construed to meet the claim language.

14 • Some of the claims recite a constraint resolver to resolve the constraint layers so
15 that operation of the application is constrained by a sum of the constraints in the layers.
16 There is absolutely no hint in Immerman that these models might conflict with each
17 other. Hence, there is likewise no suggestion that Immerman's system includes a
18 constraint resolver to cope with such conflicts.

19 The Examiner is respectfully requested to address each of these points – and to
20 particularly address these points in a manner which does not simply involve pointing to a
21 passage in Immerman. Due to the complexity and length of the Immerman document,
22 prosecution will be advanced if the Patent Office can expressly name the features of
23 Immerman that are being construed to meet the claim elements.

1 *Conclusion*

2 The arguments presented above are not exhaustive; Applicant reserves the right to
3 present additional arguments to fortify its position. Further, Applicant reserves the right
4 to challenge the alleged prior art status of one or more documents cited in the Office
5 Action.

6 All objections and rejections raised in the Office Action having been addressed,
7 it is respectfully submitted that the present application is in condition for allowance and
8 such allowance is respectfully solicited. The Examiner is urged to contact the
9 undersigned if any issues remain unresolved by this Amendment.

10

11

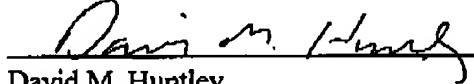
Respectfully Submitted,

12

13

14

Dated: 9-15-2005

By: 

David M. Huntley
Reg. No. 40,309
(509) 324-9256

15

16

17

18

19

20

21

22

23

24

25